REMARKS

The claims are claims 1 to 24.

The application has been amended at many locations to correct minor errors and to present uniform language throughout. The amendments include an update of the status of the co-pending application cited on page 1.

Claim 14 has been amended in response to the rejection under 35 U.S.C. 112 as suggested by the Examiner. Claims 15 and 16 have been similarly amended.

New declarations properly citing the duty to disclose under 35 U.S.C. 1.56 are attached.

Claims 1 to 12, 17 to 20, and 21 to 24 were provisionally rejected under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 1 to 4, 6, and 7 of co-pending Application No. 09/998,755. Claims 1 to 24 were provisionally rejected under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 1 to 5, 7 to 12, and 16 to 18 of co-pending Application No. 09/998,329. Claims 1 to 12, 17 to 20, and 21 to 24 were provisionally rejected under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 1 to 5, 7 to 12, and 16 to 18 of co-pending Application No. 09/998,330.

A terminal disclaimer relative to U.S. Patent Application Serial Nos. 09/998,755, 09/998,329 and 09/998,330 is attached. This terminal disclaimer obviated the obviousness-type double patenting rejection.

Claims 1, 17, and 21 were rejected under 35 U.S.C. 102(a) anticipated by Falik et al U.S. Patent No. 6,065,078.

Claims 1, 17 and 21 recite subject matter not anticipated by Falik et al. Claims 1, 17 and 21 each recite "setting a first

software breakpoint in a shared memory location in the first debug session such that all debug sessions are notified of the setting of the breakpoint." Falik et al includes no teaching regarding setting of software breakpoints. The OFFICE ACTION cites column 5, lines 17 to 21 as anticipating this subject matter. The Applicants respectfully submit that triggering an ABORT as described in this portion of Falik et al is not the same as the claimed setting of a breakpoint. Note that Falik et al states at column 5, lined 4 to 8:

> "Abort is used to stop a processor's execution flow in response to an event caused by the host 1820 (e.g., a user pressed an abort button) or another processor reached a breakpoint and a full system stop is desired to best observe the entire system status at the time of the breakpoint."

This portion of Falik et al indicates that an ABORT can result from reaching a breakpoint. Thus the portion of Falik et al cited in the OFFICE ACTION cannot teach the setting of a breakpoint. Falik et al also fails to include any teaching regarding notification of setting a software breakpoint. Note that the OFFICE ACTION includes no mention of this limitation or where Falik et al anticipates this recitation of claims 1, 17 and 24. Accordingly, claims 1, 17 and 21 are not anticipated by Falik et al.

Claims 1, 17 and 21 recite subject matter not anticipated by Falik et al. Claims 1, 17 and 21 each recite "clearing the first software breakpoint in the shared memory location in the second debug session such that all debug sessions are notified of the clearing of the breakpoint." Falik et al states at column 16, lines 36 to 48 states:

> "When one core 1840a to 1840c reaches a breakpoint, it can break all the other cores or a subset of them. The JTAG controlled ABORT mask register defines which of the processors 1840a to 1840c will be stopped at a general abort. When the

> core reaches the breakpoint it will write to the DBGABORT register a "1" to the position of all processors 1840a to 1840c it needs to stop (usually all except itself). As a result, an ISE interrupt will be sent to all the processors whose respective bit in the ABORT register is set 1840a to 1840c and their program flow will be interrupted."

This portion of Falik et al states that an ISE interrupt can result from reaching a breakpoint. Thus the ISE interrupt is not a breakpoint. The Applicants respectfully submit that clearing the ISE interrupt as described in the portion of Falik et al cited in the OFFICE ACTION fails to teach the claimed clearing of a software breakpoint. Falik et al includes no teaching regarding clearing of software breakpoints. Falik et al also fails to include any teaching regarding notification of clearing a software breakpoint. Accordingly, claims 1, 17 and 21 are not anticipated by Falik et al.

Claims 2 to 7, 18 to 20, and 22 to 24 were rejected under 35 U.S.C. 103(a) as made obvious by the combination of Falik et al U.S. Patent No. 6,065,078 and Tarui et al U.S. Patent 6,088,770.

Claims 3, 19 and 23 recite subject matter not made obvious by the combination of Falik et al and Tarui et al. Claims 3, 19 and 23 recite "searching the software memory map to find a first plurality of processors having read access to the shared memory location." The OFFICE ACTION cites Tarui et al at column 6, line 66 to column 7, line 9 and making obvious this limitation. Tarui et al states at column 6, line 66 to column 7, line 6 (within the portion cited in the OFFICE ACTION):

> "A RAT (Remote Access Table) 138 is a circuit for storing therein the attributes (whether or not a page has been accessed from any other node, and if the CCC to any other node is necessary or not) of each of the pages of the main memory included in the particular node."

This portion of Tarui et al states the remote access table stores an indication "whether or not a page has been accessed from any other node." This indication is not whether a processor has read access to a shared memory location. Thus the search disclosed in this portion of Tarui et al must yield a different result than the claimed search due to the information Tarui et al teaches is stored in the remote access table. The OFFICE ACTION includes no indication how this different information stored in Tarui et al makes obvious the search result recited in claims 3, 19 and 23. Accordingly, the combination of Falik et al and Tarui et al fail to make obvious claims 3, 19 and 23.

Claims 3, 19 and 23 recite further subject matter not made obvious by the combination of Falik et al and Tarui et al. Claims 3, 19 and 23 recite "updating a software representation maintained for software breakpoints for each of the first plurality of processors" and "writing the software breakpoint instruction in the shared memory location." The OFFICE ACTION cites Falik et al at column 16, lines 36 to 45 as making obvious both these limitations. This portion of Falik et al states:

> "When one core 1840a to 1840c reaches a breakpoint, it can break all the other cores or a subset of them. The JTAG controlled ABORT mask register defines which of the processors 1840a to 1840c will be stopped at a general abort. When the core reaches the breakpoint it will write to the DBGABORT register a '1' to the position of all processors 1840a to 1840c it needs to stop (usually all except itself). As a result, an ISE interrupt will be sent to all the processors whose respective bit in the ABORT register is set 1840a to 1840c and their program flow will be interrupted. Note that if the processor is already executing within monitor code, there will be no ISE interrupt and only the ABORT bit will be set. The monitor of each stopped processor will need to respond to the ISE interrupt the next time it polls on DBGISESRCA register. (Note that even though there is a delay in the response here, the application is not running and only monitor code is executed during this window). The purpose of this mechanism is to prevent nested calls to the monitor."

This portion of Falik et al includes no teaching of updating a software breakpoint. The only mention of a breakpoint in this portion of Falik et al refers to what happens when a breakpoint is reached. Reaching a breakpoint is not the same as the claimed updating. Note further that the updating recited in claims 3, 19 and 23 is conditional "for each of the first plurality of processors" while this portion of Falik et al includes no such limitation on processors. This portion of Falik et al fails to disclose writing to the shared memory location. The only writing disclosed in this portion of Falik et al is writing to the DBGABORT register. This is clearly not the recited shared memory. Accordingly, the combination of Falik et al and Tarui et al fail to make obvious claims 3, 19 and 23.

Claims 4, 20 and 24 recite subject matter not made obvious by the combination of Falik et al and Tarui et al. Claims 4, 20 and 24 recite "searching the software memory map for a second processor with write access to the shared memory location." The OFFICE ACTION cites Tarui et al at column 9, lines 32 to 41 as making obvious this limitation. This portion of Tarui et al discloses checking "if the access address denoted by A lies between the start address 1503 and end address 1504 of the local area of the particular node." Tarui et al states at column 9, lines 1 to 11 states:

> "When a command has been delivered onto the shared bus, the bus command reception/bus command classification circuit 131 sends an accessed address to the 'remote' decision circuit 132. The 'Remote' decision circuit 132 judges whether the delivered command is accessing the address of the main memory of the particular node of its own ('internal') or the address of the main memory of any other node ('remote'). Wherein the 'remote' decision circuit 132 makes a decision of 'internal'/'remote' by using the contents of the partition/main memory configuration information 150."

This portion of Tarui et al indicates that the check of remote decision circuit 132 is whether an accessed address is internal or remote. This determination fails to find a second processor that can access the shared memory location as recited in claims 4, 20 and 24. Because Tarui et al teaches checking for a different quality than recited in claims 4, 20 and 24. Accordingly, the combination of Falik et al and Tarui et al fail to make obvious claims 4, 20 and 24.

Claims 8 to 12 were rejected under 35 U.S.C. 103(a) as made obvious by the combination of Falik et al U.S. Patent No. 6,065,078, Tarui et al U.S. Patent 6,088,770 and Bhattacarya U.S. Patent No. 6,708,326.

Claim 9 recites subject matter not made obvious by the combination of Falik et al, Tarui et al and Bhattacarya. Claims 4, 20 and 24 recite "searching the software memory map for a third processor with write access to the shared memory location." The OFFICE ACTION cites Tarui et al at column 9, lines 32 to 41 as making obvious this limitation. This portion of Tarui et al discloses checking "if the access address denoted by A lies between the start address 1503 and end address 1504 of the local area of the particular node." As quoted above this portion of Tarui et al indicates that the check of remote decision circuit 132 is whether an accessed address is internal or remote. This determination fails to find a third processor that can access the shared memory location as recited in claim 9. Because Tarui et al teaches checking for a different quality than recited in claim 9. Accordingly, the combination of Falik et al, Tarui et al and Bhattacarya fail to make obvious claim 9.

Claims 14 to 16 were rejected under 35 U.S.C. 103(a) as made unobvious by the combination of Falik et al et al U.S. Patent No.

6,065,078, Rosenberg "How Debuggers Work," Tarui et al U.S. Patent 6,088,770.

Claims 14, 15 and 16 recite subject matter not made obvious by the combination of Falik et al, Rosenberg and Tarui et al. Claims 14, 15 and 16 each recite "searching a software memory map to find" a plurality of processors "having read access to the shared memory location." The OFFICE ACTION cites column 6, line 66 to column 7, line 7 of Tarui et al as making obvious this searching limitation. This portion of Tarui et al states the remote access table stores an indication "whether or not a page has been accessed from any other node." This indication is not whether a processor has read access to a shared memory location. Thus the search disclosed in this portion of Tarui et al must yield a different result than the claimed search due to the information Tarui et al teaches is stored in the remote access table. The OFFICE ACTION includes no indication how this different information stored in Tarui et al makes obvious the search result recited in claims 14, 15 and 16. Accordingly, the combination of Falik et al and Tarui et al fail to make obvious claims 14, 15 and 16.

The Applicants respectfully submit that all the present claims are allowable for the reasons set forth above. Therefore early reconsideration and advance to issue are respectfully requested.

If the Examiner has any questions or other correspondence regarding this application, Applicants request that the Examiner contact Applicants' attorney at the below listed telephone number and address to facilitate prosecution.

Texas Instruments Incorporated
P.O. Box 655474 M/S 3999
Dallas, Texas 75265
(972) 917-5290
Fax: (972) 917-4418

Respectfully submitted,

Robert D. Marshall, Jr.
Reg. No. 28,527